Ordinal Mind Change Complexity of Language Identification

Andris Ambainis [a], Sanjay Jain [b], Arun Sharma [c]

[a] *Department of Electrical Engineering and Computer Scienes*
Computer Science Division
University of California, Berkeley
Berkeley, CA 94720, USA
Email: ambainis@cs.berkeley.edu

[b] *Department of Information Systems and Computer Science*
National University of Singapore
Singapore 119260, Republic of Singapore
Email: sanjay@iscs.nus.edu.sg

[c] *School of Computer Science and Engineering*
The University of New South Wales
Sydney, NSW 2052, Australia
Email: arun@cse.unsw.edu.au

Abstract

The approach of ordinal mind change complexity, introduced by Freivalds and Smith, uses (notations for) constructive ordinals to bound the number of mind changes made by a learning machine. This approach provides a measure of the extent to which a learning machine has to keep revising its estimate of the number of mind changes it will make before converging to a correct hypothesis for languages in the class being learned. Recently, this notion, which also yields a measure for the difficulty of learning a class of languages, has been used to analyze the learnability of rich concept classes.

The present paper further investigates the utility of ordinal mind change complexity. It is shown that for identification from both positive and negative data and $n \geq 1$, the ordinal mind change complexity of the class of languages formed by unions of up to $n+1$ pattern languages is only $\omega \times_O notn(n)$ (where $notn(n)$ is a notation for n, ω is a notation for the least limit ordinal and $\times_O$ represents ordinal multiplication). This result nicely extends an observation of Lange and Zeugmann that pattern languages can be identified from both positive and negative data with 0 mind changes.

Existence of an ordinal mind change bound for a class of learnable languages can be seen as an indication of its learning "tractability." Conditions are investigated under which a class has an ordinal mind change bound for identification from positive data. It is shown that an indexed family of languages has an ordinal mind

Preprint submitted to Elsevier Science

change bound if it has finite elasticity and can be identified by a conservative machine. It is also shown that the requirement of conservative identification can be sacrificed for the purely topological requirement of M-finite thickness. Interaction between identification by monotonic strategies and existence of ordinal mind change bound is also investigated.

Key words: Inductive inference, mind change complexity, ordinals

1 Introduction

Natural numbers have been used as counters for bounding the number of mind changes. However, such bounds do not take into account scenarios in which a learning machine, after examining an element of the language is in a position to issue a bound on the number of mind changes it will make before the onset of convergence. For example, consider the class $COINIT = \{L \mid (\exists n)[L = \{x \mid x \geq n\}]\}$. Intuitively, $COINIT$ is the collection of languages that contain all natural numbers except a finite initial segment. Clearly, a learning machine that, at any given time, finds the minimum element n in the data seen so far and emits a grammar for the language $\{x \mid x \geq n\}$ identifies $COINIT$ in the limit from positive data. It is also easy to see that the class $COINIT$ cannot be identified by any machine that is required to converge within a constant number of mind changes. However, the machine identifying $COINIT$ can, after examining an element of the language, issue an upper bound on the number of mind changes. It turns out that the class of pattern languages ($PATTERN$), first introduced by Angluin [2] and shown to be identifiable in the limit from only positive data (texts), displays similar behavior. This is because any string in a pattern language yields a finite set of patterns that are candidate patterns for the language being learned. Such scenarios can be modeled by the use of (notations for) constructive ordinals as mind change counters introduced by Freivalds and Smith [9]. We illustrate the idea with a few examples; the formal definition is presented later.

TxtEx denotes the collection of language classes that can be identified in the limit from texts. $\mathbf{TxtEx}_\alpha$ denotes the collection of language classes that can be identified in the limit from texts with an ordinal mind change bound α. Let ω denote a notation for the least limit ordinal. For $\alpha \prec \omega$, the notion coincides with the earlier notion of bounded mind change identification [5,7]. For $\alpha = \omega$, $\mathbf{TxtEx}_\omega$ denotes learnable classes for which there exists a machine that, after examining some element(s) of the language, can announce an upper bound on the number of mind changes it will make before the onset of successful convergence. Both, $COINIT$ and $PATTERN$ are members of $\mathbf{TxtEx}_\omega$. Let $notn(n)$ denote an ordinal notation for natural number n and let $\times_O$ represent ordinal multiplication. Proceeding on, the class $\mathbf{TxtEx}_{\omega \times_O notn(2)}$ contains classes for which there is a learning machine that after examining some element(s) of the language announces an upper bound on the number of mind changes, but reserves the right to revise this upper bound once. Similarly, in the case of $\mathbf{TxtEx}_{\omega \times_O notn(3)}$, the machine reserves the right to revise its upper bound twice, and so on. $\mathbf{TxtEx}_{\omega \times_O \omega}$ contains classes for which the machine announces an upper bound on the number of times it may revise its conjectured upper bound on the number of mind changes, and so on.

Shinohara [31] showed that the class of pattern languages is not closed under union and many rich concepts can be represented by unions of pattern languages; these languages have been applied to knowledge acquisition from amino acid sequences (see Arikawa et al. [4]). For empirical approaches to learning unions of simple pattern languages, see Kilpeläinen, Mannila, and Ukkonen [18]. In [12,14], the ordinal mind change complexity of the classes of languages formed by taking unions of pattern languages was derived. For $n \geq 1$, it was shown that the class formed by taking unions of up to n pattern languages, $PATTERN^n$, is in $\mathbf{TxtEx}_{\omega^n}$, where ω^n denotes $\omega \times_O \omega \ldots \times_O \omega$ (ω is multiplied by itself n times). It was also shown that there are cases for which the ω^n bound is essential because $PATTERN^n \notin \mathbf{TxtEx}_\alpha$, for all $\alpha \prec \omega^n$.

In this paper we investigate the ordinal mind change bounds for identification in the limit of unions of pattern languages from both positive and negative data (informants). **InfEx** denotes the collection of language classes that can be identified in the limit from informants and $\mathbf{InfEx}_\alpha$ denotes the collection of those classes identifiable with an ordinal mind change bound of α. Lange and Zeugmann [20] have observed that *PATTERN* can be identified from informants with 0 mind changes. So, it is to be expected that the ordinal mind change bounds for identification from informants of unions of pattern languages be lower than those for identification from texts. We show that this is indeed the case as, for $n \geq 1$, $PATTERN^{n+1} \in \mathbf{InfEx}_{\omega \times_O notn(n)}$.

It is interesting to note that although the unbounded union of pattern languages is not identifiable from texts, it is identifiable from informants. Unfortunately, there is no ordinal mind change bound for identification from informants of unbounded unions of pattern languages. This is because this class contains the class of finite languages, *FIN*, for which there is no ordinal mind change complexity bound. It may be argued that in terms of mind change complexity, *FIN* is a very difficult problem. [1] Since the existence of ordinal mind change bound for a class is a reflection of its learning "tractability", it is therefore useful to investigate conditions under which an ordinal

[1] A similar conclusion can be drawn from the study of intrinsic complexity of *FIN* [11,13,15], where it turns out that *FIN* is a complete class with respect to weak reduction.

mind change bound can be guaranteed. We consider a number of possibilities, including identification by conservative strategies, topological properties like finite thickness, M-finite thickness, and finite elasticity, and monotonicity requirements. We preview some of our results.

We first establish a useful technical result which states that if a learning machine makes a finite number of mind changes on any text, then the class of languages that can be identified by this machine has an ordinal mind change bound. This result is used to show that if an indexed family of languages has finite elasticity and can be conservatively identified then there is an ordinal mind change bound for this class. We also show that the requirement of conservative identification can be sacrificed in the previous result for the purely topological requirement that the class have M-finite thickness in addition to finite elasticity. Since finite thickness implies finite elasticity and M-finite thickness, the above results imply that any indexed family of languages with finite thickness has an ordinal mind change bound.

The results discussed above give general sufficient conditions for identifiability with ordinal bound on mind changes. However, the mind change bound α may be arbitrarily large. An interesting question to ask is whether the ordinal mind change bound remains arbitrarily large if some other constraints such as monotonicity are added. We show a negative result in this direction as for every constructive ordinal bound α, there exists an indexed family of languages that can be identified strong-monotonically and has finite thickness, but cannot be identified with the ordinal mind change bound of α. A similar result also holds for dual strong-monotonicity.

We now proceed formally.

2 Preliminaries

N denotes the set of natural numbers, $\{0, 1, 2, \ldots\}$; Any unexplained recursion theoretic notation is from [27]. Cardinality of a set S is denoted $\mathrm{card}(S)$. The maximum and minimum of a set are represented by $\max(\cdot)$ and $\min(\cdot)$, respectively. The symbols $\subseteq, \supseteq, \subset, \supset$, and $\emptyset$ respectively stand for subset, superset, proper subset, proper superset, and the emptyset. A language is any subset of N. L is a typical variable for a language. $\overline{L}$ is the complement of L, that is, $\overline{L} = N - L$.

We first define the notion of texts for languages.

Definition 1 *[10]*

(a) *A* text *T is a mapping from N into $N \cup \{\#\}$.*
(b) *A text T is for a language L iff L is the set of natural numbers in the range of T.*
(c) content(T) *denotes the set of natural numbers in the range of T.*
(d) *The initial sequence of text T of length n is denoted $T[n]$.*
(e) *The set of all finite initial sequences of N and $\#$'s is denoted SEQ.*

Members of SEQ are inputs to machines that learn grammars (acceptors) for r.e. languages. We let σ and τ, with or without decorations [2], range over SEQ. Λ denotes the empty sequence. content(σ) denotes the set of natural numbers in the range of σ and the length of σ is denoted $|\sigma|$. We say that $\sigma \subseteq \tau$ ($\sigma \subseteq T$) to denote that σ is an initial sequence of τ (T).

Definition 2 *A language learning machine (from texts) is an algorithmic mapping from SEQ into $N \cup \{?\}$.*

A conjecture of "?" by a machine is interpreted as "no guess at this moment." This is useful to avoid biasing the number of mind changes of a machine. For this paper, we assume, without loss of generality, that $\sigma \subseteq \tau$ and $\mathbf{M}(\sigma) \neq ?$ implies $\mathbf{M}(\tau) \neq ?$.

$\mathbf{M}$ denotes a typical variable for a language learning machine (from texts or informants). We also fix an acceptable programming system [22] and interpret the output of a language learning machine as the index of a program in this system. We associate these programs with the domain of the partial functions computed by them. Then, a program conjectured by a machine in response to a finite initial sequence may be viewed as a candidate accepting grammar for the language being learned. We say that $\mathbf{M}$ converges on text T to i (written: $\mathbf{M}(T)$ converges to i or $\mathbf{M}(T)\!\downarrow = i$) just in case for all but finitely many n, $\mathbf{M}(T[n]) = i$. The following definition introduces Gold's criterion for successful identification of languages.

Definition 3 *[10]*

(a) $\mathbf{M}$ **TxtEx**-identifies *a text T just in case $\mathbf{M}(T)$ converges to a grammar for* content(T).

[2] Decorations are subscripts, superscripts and the like.

(b) $\mathbf{M}$ **TxtEx**-identifies *an r.e. language L (written: $L \in \mathbf{TxtEx}(\mathbf{M})$) just in case* $\mathbf{M}$ **TxtEx**-*identifies each text T for L.*
(c) $\mathbf{M}$ **TxtEx**-identifies *a class $\mathcal{L}$ of r.e. langauges, iff* $\mathbf{M}$ **TxtEx**-*identifies each $L \in \mathcal{L}$.*
(d) **TxtEx** *denotes the family of all sets $\mathcal{C}$ of r.e. languages such that some machine* **TxtEx**-*identifies each language in $\mathcal{C}$.*

The next two definitions describe the notion of informants as a model of both positive and negative data presentation and identification in the limit from informants.

Definition 4 *[10]*

(a) *An* informant *I is an infinite sequence over $N \times \{0,1\}$ such that for each $n \in N$ either $(n,1)$ or $(n,0)$ (but not both) appear in the sequence.*
(b) *An informant I is for L iff $(n,1)$ appears in I if $n \in L$ and $(n,0)$ appears in I if $n \notin L$.*
(c) *$I[n]$ denotes the initial sequence of informant I with length n.*
(d) $\mathrm{content}(I) = \{(x,y) \mid (x,y)$ *appears in sequence* $I\}$. $\mathrm{content}(I[n])$ *is defined similarly.*
(e) $\mathrm{PosInfo}(I[n]) = \{x \mid (x,1) \in \mathrm{content}(I[n])\}$. $\mathrm{NegInfo}(I[n]) = \{x \mid (x,0) \in \mathrm{content}(I[n])\}$.
(f) $SEG = \{I[n] \mid I$ *is an informant for some* $L \subseteq N\}$.

A language learning machine (from informants) is an algorithmic mapping from SEG into $N \cup \{?\}$. We say that $\mathbf{M}$ converges on informant I to i (written: $\mathbf{M}(I)$ converges to i or $\mathbf{M}(I)\!\downarrow = i$) just in case for all but finitely many n, $\mathbf{M}(I[n]) = i$.

We now define identification from informants.

Definition 5 *[10]*

(a) $\mathbf{M}$ **InfEx**-identifies *an r.e. language L just in case* $\mathbf{M}$*, fed any informant for L, converges to a grammar for L. In this case we say that $L \in \mathbf{InfEx}(\mathbf{M})$.*
(b) $\mathbf{M}$ **InfEx**-identifies *a collection of languages, $\mathcal{C}$, just in case* $\mathbf{M}$ **InfEx**-*identifies each language in $\mathcal{C}$.*
(c) **InfEx** *denotes the family of all sets $\mathcal{C}$ of r.e. languages such that some machine* **InfEx**-*identifies $\mathcal{C}$.*

The following proposition describes the relationship between **TxtEx** and **InfEx**.

Proposition 6 [10] $\mathbf{TxtEx} \subset \mathbf{InfEx}$.

We assume a fixed notation system, O, and partial ordering of ordinal notations as used by, for example, Kleene [19,27,28]. $\preceq, \prec, \succeq$ and $\succ$ on ordinal notations below refer to the partial ordering of ordinal notations in this system. Similarly, $\times_O$ and $+_O$ refer to the addition and multiplication of the ordinal notations in this system. We do not go into the details of the notation system used, but instead refer the reader to [19,27,28,6,9].

For a natural number n, we let $notn(n)$ denote a notation for n. We let ω denote a notation for the least limiting ordinal.

Definition 7 $\mathbf{F}$*, an algorithmic mapping from SEQ (or SEG) into ordinal notations, is an* ordinal mind change counter function *just in case* $(\forall \sigma \subseteq \tau)[\mathbf{F}(\sigma) \succeq \mathbf{F}(\tau)]$.

Definition 8 *[9] Let α be an ordinal notation.*

(a) *We say that* $\mathbf{M}$*, with associated ordinal mind change counter function* $\mathbf{F}$*,* $\mathbf{TxtEx}_\alpha$-identifies *a text T just in case the following three conditions hold:*
  (i) $\mathbf{M}(T)$ *converges to a grammar for* content(T),
  (ii) $\mathbf{F}(\Lambda) = \alpha$ *and*
  (iii) $(\forall n)[? \neq \mathbf{M}(T[n]) \neq \mathbf{M}(T[n+1]) \Rightarrow \mathbf{F}(T[n]) \succ \mathbf{F}(T[n+1])]$.

(b) $\mathbf{M}$*, with associated ordinal mind change counter function* $\mathbf{F}$*,* $\mathbf{TxtEx}_\alpha$-identifies L *(written: $L \in \mathbf{TxtEx}_\alpha(\mathbf{M}, \mathbf{F})$) just in case* $\mathbf{M}$*, with associated ordinal mind change counter function* $\mathbf{F}$*,* $\mathbf{TxtEx}_\alpha$*-identifies each text for* L.

(c) $\mathbf{TxtEx}_\alpha = \{\mathcal{C} \mid (\exists \mathbf{M}, \mathbf{F})[\mathcal{C} \subseteq \mathbf{TxtEx}_\alpha(\mathbf{M}, \mathbf{F})]\}$.

Definition 9 *[9] Let α be an ordinal notation.*

(a) *We say that* $\mathbf{M}$*, with associated ordinal mind change counter function* $\mathbf{F}$*,* $\mathbf{InfEx}_\alpha$-identifies *an informant I for a language L just in case the following three conditions hold:*
  (i) $\mathbf{M}(I)$ *converges to a grammar for L,*
  (ii) $\mathbf{F}(\Lambda) = \alpha$ *and*
  (iii) $(\forall n)[? \neq \mathbf{M}(I[n]) \neq \mathbf{M}(I[n+1]) \Rightarrow \mathbf{F}(I[n]) \succ \mathbf{F}(I[n+1])]$.

(b) $\mathbf{M}$*, with associated ordinal mind change counter function* $\mathbf{F}$*,* $\mathbf{InfEx}_\alpha$-identifies L *(written: $L \in \mathbf{InfEx}_\alpha(\mathbf{M}, \mathbf{F})$) just in case* $\mathbf{M}$*, with associated ordinal mind change counter function* $\mathbf{F}$*,* $\mathbf{InfEx}_\alpha$*-identifies each informant for L.*

(c) $\mathbf{InfEx}_\alpha = \{\mathcal{C} \mid (\exists \mathbf{M}, \mathbf{F})[\mathcal{C} \subseteq \mathbf{InfEx}_\alpha(\mathbf{M}, \mathbf{F})]\}$.

We refer the reader to Ambainis [1] for a discussion on how the learnability classes depend on the choice of the ordinal notation.

We now formally show that $COINIT \in \mathbf{TxtEx}_\omega$. To see this, for $n \in N$, let i_n denote a grammar, obtained effectively from n, for the language $\{x \mid x \geq n\}$. We define a learning machine $\mathbf{M}$ and an ordinal mind change counter function $\mathbf{F}$ on text T as follows.

$$\mathbf{M}(T[n]) = \begin{cases} ?, & \text{if content}(T[n]) = \emptyset; \\ i_{\min(\text{content}(T[n]))}, & \text{otherwise.} \end{cases}$$

$$\mathbf{F}(T[n]) = \begin{cases} \omega, & \text{if content}(T[n]) = \emptyset; \\ notn(m), & \text{if content}(T[n]) \neq \emptyset, \text{ and } m = \min(\text{content}(T[n])). \end{cases}$$

It is easy to verify that $COINIT \subseteq \mathbf{TxtEx}(\mathbf{M}, \mathbf{F})$.

The following Lemma is useful in proving some of our theorems.

Lemma 10 *Fix an ordinal notation α. There exists an r.e. sequence of pairs of learning machines and corresponding ordinal mind change counter functions,* $(\mathbf{M}_0, \mathbf{F}_0), (\mathbf{M}_1, \mathbf{F}_1), \ldots$, *such that*

(a) for all $\mathcal{C} \in \mathbf{TxtEx}_\alpha$, there exists an i such that $\mathcal{C} \subseteq \mathbf{TxtEx}_\alpha(\mathbf{M}_i, \mathbf{F}_i)$.
(b) for all i, $\mathbf{F}_i(\Lambda) = \alpha$.
(c) for all i, for all texts T, for all n, $\mathbf{M}_i(T[n]) \neq \mathbf{M}_i(T[n+1]) \Rightarrow \mathbf{F}_i(T[n]) \succ \mathbf{F}_i(T[n+1])$.

The above lemma can be proved on the lines of the proof of Lemma 4.2.2B in [25].

3 Ordinal Mind Change Complexity of Unions of Pattern Languages

Let Σ and X be mutually disjoint sets. Σ is finite and its elements are referred to as *constant symbols*. X is countably infinite and its elements are referred to as *variables*. For the present section, we let $a, b, \ldots$ range over constant symbols and $x, y, z, x_1, x_2, \ldots$ range over variables. For a set A, let A^* denote the set of all the finite strings over A, and A^+ denote the set of all non-empty finite strings over A.

Definition 11 *[2] A* pattern *is an element of* $(\Sigma \cup X)^+$. *A* string *is an element of* Σ^+.

A *substitution* is a homomorphism from patterns to patterns that maps each symbol $a \in \Sigma$ to itself. The image of a pattern p under a substitution θ is denoted $p\theta$. We next describe the language defined by a pattern. Note that there exists a recursive bijective mapping between elements of Σ^+ and N. Thus we can name elements of Σ^+ with elements of N. We implicitly assume such a mapping when we discuss languages defined using subsets of Σ^+ below. (We do not explicitly use such a bijective mapping for ease of notation.)

Definition 12 *[2] The language associated with the pattern p is defined as $\mathbf{Lang}(p) = \{p\theta \mid \theta$ is a substitution and $p\theta \in \Sigma^+\}$. We define the class $PATTERN = \{\mathbf{Lang}(p) \mid p$ is a pattern$\}$.*

Angluin [2] showed that $PATTERN \in \mathbf{TxtEx}$. Shinohara [31] showed that pattern languages are not closed under union, and hence it is useful to study identification of languages that are unions of more than one pattern language, as they can be used to represent more expressive concepts. We next define unions of pattern languages.

Let S be a set of patterns. Then $\mathbf{Lang}(S)$ is defined as $\bigcup_{p \in S} \mathbf{Lang}(p)$. Intuitively, $\mathbf{Lang}(S)$ is the language formed by the union of languages associated with the patterns in S.

Definition 13 *[31,33] Let $n \in N$. $PATTERN^n = \{\mathbf{Lang}(S) \mid 0 < \mathrm{card}(S) \leq n\}$.*

Shinohara [31] and Wright [33] showed that for $n > 1$, $PATTERN^n \in \mathbf{TxtEx}$. Jain and Sharma [14] showed that $PATTERN^n \in \mathbf{TxtEx}_{\omega^n}$ and $PATTERN^n \notin \mathbf{TxtEx}_\alpha$ for $\alpha \prec \omega^n$.

We now consider the ordinal mind change complexity of identifying unions of pattern languages from informants. A pattern is *canonical* [2] iff it satisfies the following: if k is the number of variables appearing in a pattern p, then the variables occurring in p are precisely $\{x_1, x_2, \ldots, x_k\}$, and, for every i, $1 \leq i < k$, the leftmost occurrence of x_i in p, is to the left of the leftmost occurrence of x_{i+1} in p. Let PAT denote the set of all canonical patterns. Let $PAT^n = \{S \mid S \subseteq PAT \ \wedge \ 0 < \mathrm{card}(S) \leq n\}$.

Angluin showed that, for $p, p' \in PAT$, $\mathbf{Lang}(p) = \mathbf{Lang}(p')$ iff $p = p'$. This result does not hold for elements of PAT^n where $n > 1$.

Suppose Pos and Neg are disjoint finite sets such that $\mathrm{Pos} \neq \emptyset$. Then let

$$X_i^{\mathrm{Pos},\mathrm{Neg}} = \{S \in PAT^i \mid [\mathrm{Pos} \subseteq \mathbf{Lang}(S)] \ \wedge \ [\mathrm{Neg} \subseteq \overline{\mathbf{Lang}(S)}]\}$$

Lemma 14 *Suppose we are given finite disjoint sets* Pos, Neg*, where* $\mathrm{Pos} \neq \emptyset$*, and a natural number i, such that $(\forall j \leq i)[X_j^{\mathrm{Pos},\mathrm{Neg}} = \emptyset]$. Then, effectively in*

Pos, Neg, *and* i, *we can determine* $X_{i+1}^{\mathrm{Pos,Neg}}$. *(Note that* $X_{i+1}^{\mathrm{Pos,Neg}}$ *must be finite in this case!)*

PROOF. Suppose Pos, Neg, and i are as given in the hypothesis of the lemma. Let

$$P = \{p \in PAT \mid [\mathrm{Pos} \cap \mathbf{Lang}(p) \neq \emptyset] \ \wedge \ [\mathrm{Neg} \cap \mathbf{Lang}(p) = \emptyset]\}$$

Let

$$X = \{S \in PAT^{i+1} \mid [\mathrm{Pos} \subseteq \mathbf{Lang}(S)] \ \wedge \ [S \subseteq P]\}$$

It is easy to verify that $X = X_{i+1}^{\mathrm{Pos,Neg}}$. Also note that X can be obtained effectively from Pos, Neg and i. $\square$

Corollary 15 *Suppose* Pos *and* Neg *are disjoint finite sets such that* Pos $\neq \emptyset$. *Then effectively in* Pos, Neg, *one can find* i, *and corresponding* $X_i^{\mathrm{Pos,Neg}}$ *(which must be finite) such that* $i = \min(\{j \mid X_j^{\mathrm{Pos,Neg}} \neq \emptyset\})$.

PROOF. Note that PAT^0 is empty. The corollary now follows by repeated use of Lemma 14, until one finds an i such that $X_i^{\mathrm{Pos,Neg}} \neq \emptyset$. $\square$

Theorem 16 *(a)* $PATTERN \in \mathbf{InfEx}_{notn(0)}$.

(b) $(\forall i \geq 1)[PATTERN^{i+1} \in \mathbf{InfEx}_{\omega \times_O notn(i)}]$.

PROOF. (a) Shown by Lange and Zeugmann [20]. Also follows from the proof of Part (b).
(b) Fix i. For Z, a finite subset of PAT, let G_Z denote a grammar (obtained effectively from Z) for $\mathbf{Lang}(Z)$. Let $\mathbf{M}(I[n]), \mathbf{F}(I[n])$ be defined as follows.

Let Pos $=$ PosInfo$(I[n])$ and Neg $=$ NegInfo$(I[n])$.

If Pos $= \emptyset$, then $\mathbf{M}(I[n]) =?$ and $\mathbf{F}(I[n]) = \omega \times_O notn(i)$.

If Pos $\neq \emptyset$, then let $j = \min(\{j' \mid X_{j'}^{\mathrm{Pos,Neg}} \neq \emptyset\})$. Note that j (and corresponding $X_j^{\mathrm{Pos,Neg}}$) can be found effectively in $I[n]$, using Corollary 15.

If $j = 1$ and $\mathrm{card}(X_j^{\mathrm{Pos,Neg}}) > 1$, then $\mathbf{M}(I[n]) =?$, and $\mathbf{F}(I[n]) = \omega \times_O notn(i)$.

If $j > 1$ or $\mathrm{card}(X_j^{\mathrm{Pos,Neg}}) = 1$, then $\mathbf{M}(I[n]) = G_Z$, where Z is the lexicographically least element in $X_j^{\mathrm{Pos,Neg}}$, and $\mathbf{F}(I[n]) = \omega \times_O notn(k) +_O notn(\ell)$, where $k = i + 1 - j$, and $\ell = \mathrm{card}(X_j^{\mathrm{Pos,Neg}}) - 1$.

It is easy to verify that $\mathbf{M}, \mathbf{F}$ witness the theorem. $\square$

It is open at this stage whether we can do better than the $\omega \times_O notn(i)$ bound for $PATTERN^{i+1}$. However, if we consider unions of $i+1$ simple pattern languages [3] , then it is easy to see that the mind change bound for identification from informants is simply i.

4 Ordinal Complexity and Conservativeness

We first establish an important technical result.

Theorem 17 *Let* $\mathbf{M}$ *be a learning machine such that for any text* T *(irrespective of whether* $\mathbf{M}$ *identifies* T *or not),* $\mathbf{M}$ *makes only finitely many mind changes on* T *as input. Let* $\mathcal{C}$ *denote the class of all languages* **TxtEx**-*identified by* $\mathbf{M}$. *Then, for some ordinal mind change counter function* $\mathbf{F}$*, and constructive ordinal notation* α*,* $\mathcal{C} \subseteq \mathbf{TxtEx}_\alpha(\mathbf{M}, \mathbf{F})$.

PROOF. We define a *conjecture tree* $\mathcal{T}_\mathbf{M}$ for machine $\mathbf{M}$. The root of $\mathcal{T}_\mathbf{M}$ corresponds to the empty sequence, Λ. Other nodes of the tree correspond to finite initial sequences of texts, $T[n+1]$, such that $\mathbf{M}(T[n]) \neq \mathbf{M}(T[n+1])$. Let $S = \{\Lambda\} \cup \{T[n+1] \mid n \in N, T \text{ is a text and } \mathbf{M}(T[n]) \neq \mathbf{M}(T[n+1])\}$. For $\sigma \in S$, we use V_σ to denote the node corresponding to the sequence σ. Node V_{σ_1} is a descendent of node V_{σ_2} iff $\sigma_2 \subset \sigma_1$.

We will now define a constructive ordinal notation, α_σ, corresponding to each $\sigma \in S$. For $\sigma \in S$, let $S_\sigma = \{\tau \in S \mid \sigma \subset \tau\}$. Intuitively, S_σ denotes the proper descendants of σ in the tree $\mathcal{T}_\mathbf{M}$. Note that S_σ is recursively enumerable (effectively in σ). Let S_σ^s denote the finite set enumerated in s steps in some, effective in σ, enumeration of S_σ.

α_σ is defined as follows. α_σ is the limit of $f_\sigma(0), f_\sigma(1), \ldots$, where f_σ is defined as follows.

$f_\sigma(0) = notn(0)$. $f_\sigma(i+1) = f_\sigma(i) +_O \alpha_{\tau_1} +_O \ldots +_O \alpha_{\tau_k} +_O notn(1)$, where $\tau_1, \tau_2, \ldots, \tau_k$ are the elements of S_σ^i.

We first need to show that the α_σ's constitute a correct notation.

Lemma 18 *(a) Let* V_σ *be a leaf of* $\mathcal{T}_\mathbf{M}$. *Then* α_σ *is a correct ordinal notation.*

(b) Suppose $\sigma \in S$*, and* α_τ *is a correct ordinal notation for each* $\tau \in S_\sigma$. *Then* α_σ *is a correct ordinal notation.*

[3] A simple pattern language is formed by substituting, for each variable, strings of length exactly one.

(c) For any $\sigma \in S$, α_σ is a correct ordinal notation.

(d) If $\sigma \in S$ and $\tau \in S_\sigma$, then $\alpha_\tau \prec \alpha_\sigma$.

PROOF. (a) If V_σ is a leaf, then S_σ is empty. Hence, $f_\sigma(n) = notn(n)$. It follows that α_σ is a notation for ω.

(b) Since, α_σ is a limit of $f_\sigma(0), f_\sigma(1), \ldots$, it suffices to show that each $f_\sigma(i)$ is a correct ordinal notation. Now, for each $\tau \in S_\sigma$, α_τ is a correct notation. Thus, since $f_\sigma(i+1)$ is defined using $f_\sigma(i)$, α_τ, $notn(1)$ and $+_O$ operation only, $f_\sigma(i+1)$ is a correct ordinal notation.

(c) Suppose by way of contradiction that α_σ is not a correct notation. We then construct an infinite sequence $\sigma_0 \subset \sigma_1 \subset \ldots$ such that, for each i, $\sigma_i \in S$ and α_{σ_i} is not a correct notation.

Let $\sigma_0 = \sigma$. Suppose σ_i has been defined. Let σ_{i+1} be such that $\sigma_{i+1} \in S_{\sigma_i}$ and $\alpha_{\sigma_{i+1}}$ is not a correct notation. The existence of such a σ_{i+1} follows from parts (a) and (b).

Consider the text $T = \bigcup_{i \in N} \sigma_i$. Now, since each $\sigma_i \in S$, we have that $\mathbf{M}$ on T makes infinitely many mind changes (after reading last element of σ_1, after reading last element of σ_2, and so on). This yields a contradiction to the hypothesis of the theorem.

(d) Note that $\alpha_\sigma \succ f_\sigma(i)$, for each i. Suppose $\tau \in S_\sigma^s$. Then it is easy to see that $f_\sigma(s+1) \succ \alpha_\tau$. Thus $\alpha_\tau \prec \alpha_\sigma$. $\Box$

We continue with the proof of the theorem. Let $\alpha = \alpha_\Lambda$. We now construct an $\mathbf{F}$ such that $\mathcal{C} \subseteq \mathbf{TxtEx}_\alpha(\mathbf{M}, \mathbf{F})$. $\mathbf{F}$ is defined as follows.

$$\mathbf{F}(T[n]) = \begin{cases} \alpha_\Lambda, & \text{if } T[n] = \Lambda; \\ \mathbf{F}(T[n]-1), & \text{if } n > 0, \text{ and } \mathbf{M}(T[n+1]) = \mathbf{M}(T[n]); \\ \alpha_{T[n]}, & \text{otherwise.} \end{cases}$$

From the definition of α_σ and Lemma 18, it is easy to verify that $\mathbf{TxtEx}(\mathbf{M}) \subseteq \mathbf{TxtEx}_\alpha(\mathbf{M}, \mathbf{F})$. $\Box$

Theorem 17 allows us to establish several sufficient conditions for the existence of ordinal bounds on mind changes in the context of identification of indexed families of languages. We first adapt learnability notions to the context of indexed families of languages.

A sequence of nonempty languages $\mathcal{L} = L_0, L_1, \ldots$ is an indexed family of languages (sometimes called just indexed family) if there exists a computable function f such that for each $i \in N$ and for each $x \in N$,

$$f(i, x) = \begin{cases} 1, & \text{if } x \in L_i, \\ 0, & \text{otherwise.} \end{cases}$$

In other words, there is a uniform decision procedure for languages in the family. Here, i may be thought of as a grammar for the language L_i. In the sequel, we let $\mathcal{L}$, with or without decorations, range over indexed families. For an indexed family $\mathcal{L} = L_0, L_1, \ldots$, we let $\mathrm{range}(\mathcal{L}) = \{L_i \mid i \in N\}$. For learning indexed families, usually one considers indexed families as hypothesis spaces [21]. The next definition adapts Gold's criterion of identification in the limit to the identification of indexed families with respect to a given hypothesis space.

Definition 19 *[10,3] Let $\mathcal{L}$ be an indexed family and let $\mathcal{L}' = L'_0, L'_1, \ldots$ be a hypothesis space.*

(a) Let $L \in \mathrm{range}(\mathcal{L})$. A machine $\mathbf{M}$ $\mathbf{TxtEx}$-identifies L *with respect to (hypothesis space) $\mathcal{L}'$ just in case for any text T for L, there exists j such that $\mathbf{M}(T){\downarrow} = j$ and $L = L'_j$.*

(b) A machine $\mathbf{M}$ $\mathbf{TxtEx}$-identifies $\mathcal{L}$ *with respect to $\mathcal{L}'$ just in case for each $L \in \mathrm{range}(\mathcal{L})$,* $\mathbf{M}$ $\mathbf{TxtEx}$*-identifies L with respect to $\mathcal{L}'$.*

There are three kinds of identification that have been studied in the literature: (a) class comprising; (b) class preserving; and (c) exact. If the indexed family $\mathcal{L}$ is identified with respect to a hypothesis space $\mathcal{L}'$ such that $\mathrm{range}(\mathcal{L}) \subseteq \mathrm{range}(\mathcal{L}')$ then the identification is referred to as *class comprising*. However, if it is required that the indexed family be identifiable with respect to a hypothesis space $\mathcal{L}'$ such that $\mathrm{range}(\mathcal{L}) = \mathrm{range}(\mathcal{L}')$ then the identification is referred to as *class preserving*. Finally, if the identification of the indexed family $\mathcal{L}$ is required to be with respect to $\mathcal{L}$ itself, then the identification is referred to as *exact*. The reader is directed to the excellent survey by Zeugmann and Lange [34] for discussion of these issues.

We can similarly define $\mathbf{TxtEx}_\alpha$-identification with respect to hypothesis space $\mathcal{L}'$. Note that Theorem 17 holds with respect to all hypothesis spaces.

We next describe certain topological conditions on language classes that yield sufficient conditions for identifiability of indexed families. The following notion was introduced by Angluin [2].

Definition 20 *[2] $\mathcal{L}$ has* finite thickness *just in case for each $n \in N$,* $\mathrm{card}(\{L \in \mathrm{range}(\mathcal{L}) \mid n \in L\})$ *is finite.*

Angluin [2] showed that if $\mathcal{L}$ is an indexed family and $\mathcal{L}$ has finite thickness then $\mathcal{L} \in \mathbf{TxtEx}$. A more interesting topological notion was introduced by Wright [33] (see also Motoki, Shinohara, and Wright [23]) described below.

Definition 21 *[33,23] $\mathcal{L}$ has* infinite elasticity *just in case there exists an infinite sequence of pairwise distinct numbers, $\{w_i \in N \mid i \in N\}$, and an infinite sequence of pairwise distinct languages, $\{A_i \in \text{range}(\mathcal{L}) \mid i \in N\}$, such that for each $k \in N$, $\{w_i \mid i < k\} \subseteq A_k$, but $w_k \notin A_k$. $\mathcal{L}$ is said to have* finite elasticity *just in case $\mathcal{L}$ does not have infinite elasticity.*

Wright [33] showed that if $\mathcal{L}$ has finite thickness then it has finite elasticity. He further showed that if $\mathcal{L}$ is an indexed family and $\mathcal{L}$ has finite elasticity, then $\mathcal{L} \in \mathbf{TxtEx}$.

Finite elasticity is a sufficient condition for identification of indexed families. Also, the property of finite elasticity is preserved under finite unions. As already noted, it was shown in [14] that for each $n > 0$, $PATTERN^n \in \mathbf{TxtEx}_{\omega^n}$. It would be interesting to investigate whether, for each indexed family $\mathcal{L}$ that has finite elasticity, there is an i such that $\mathcal{L} \in \mathbf{TxtEx}_{\omega^i}$. The following result established in [14] showed that the answer to this question is negative.

Theorem 22 [14] *There exists an indexed family, $\mathcal{L}$, such that (a) $\mathcal{L}$ has finite elasticity and (b) for each $i > 0$, $\mathcal{L} \notin \mathbf{TxtEx}_{\omega^i}$.*

However, we are able to show that an indexed family with finite elasticity has an ordinal mind change bound if it can be identified conservatively. The next definition describes conservative identification.

Definition 23 *[3] Let $\mathcal{L} = L_0, L_1, \ldots$ be a hypothesis space. $\mathbf{M}$ is said to be a* conservative learning machine with respect to the hypothesis space $\mathcal{L}$ *just in case for all σ and τ such that $\sigma \subseteq \tau$ and $\text{content}(\tau) \subseteq L_{\mathbf{M}(\sigma)}$, $\mathbf{M}(\sigma) = \mathbf{M}(\tau)$.*

Intuitively, conservative machines do not change their hypothesis if the input is contained in the language conjectured.

Theorem 24 *Let $\mathcal{L}' = L'_0, L'_1, \ldots$ be an indexed family with finite elasticity. Assume that $\mathcal{L}$ is identifiable by a conservative learning machine with respect to the hypothesis space $\mathcal{L}'$. Then $\mathcal{L} \in \mathbf{TxtEx}_\alpha$ with respect to hypothesis space $\mathcal{L}'$, for some constructive ordinal notation α.*

PROOF. Let $\mathbf{M}$ be a conservative learning machine which identifies $\mathcal{L}$ with respect to hypothesis space $\mathcal{L}'$. We will describe a machine $\mathbf{M}'$ which identifies $\mathcal{L}$ with respect to $\mathcal{L}'$, and changes its mind at most finitely often on every text. Theorem 17 will then imply the theorem.

For a given text T, $n \in N$, let $\text{lmc}(\mathbf{M}', T[n])$ be defined as follows:

$$\text{lmc}(\mathbf{M}', T[n]) = \max(\{m+1 \mid m < n \ \wedge \ \mathbf{M}'(T[m]) \neq \mathbf{M}'(T[m+1])\})$$

Intuitively, lmc denotes the last point where $\mathbf{M}'$ made a mind change. Note that if $\mathbf{M}'(T[0]) = \mathbf{M}'(T[1]) = \cdots = \mathbf{M}'(T[n])$, then $\text{lmc}(\mathbf{M}', T[n]) = 0$. $\mathbf{M}'$ is now defined as follows:

$$\mathbf{M}'(T[n]) = \begin{cases} ?, & \text{if } n = 0 \text{ or } \mathbf{M}(T[n]) = ?; \\ \mathbf{M}(T[n]), & \text{if } \text{content}(T[\text{lmc}(\mathbf{M}', T[n-1])]) \subseteq L'_{\mathbf{M}(T[n])}; \\ \mathbf{M}'(T[n-1]), & \text{otherwise.} \end{cases}$$

It is easy to verify that $\mathbf{M}'$ **TxtEx**-identifies with respect to $\mathcal{L}'$ any language which $\mathbf{M}$ **TxtEx**-identifies with respect to $\mathcal{L}'$. We prove that $\mathbf{M}'$ makes only finitely many mind changes on any text T. By Theorem 17, this implies that $\mathcal{L} \in \mathbf{TxtEx}_\alpha$ with respect to hypothesis space $\mathcal{L}'$, for some constructive ordinal notation α.

Suppose by way of contradiction that $\mathbf{M}'$ makes infinitely many mind changes on a text T. Let $n_1 < n_2 < \ldots$ be such that, for each i, $\mathbf{M}'(T[n_i]) \neq \mathbf{M}'(T[n_i + 1])$. Then, it is easy to verify from the construction of $\mathbf{M}'$ that, for all i, $\text{content}(T[n_i + 1]) \subseteq L'_{\mathbf{M}'(T[n_{i+2}])}$. Moreover, since $\mathbf{M}$ is conservative, we have $\text{content}(T[n_i + 1]) \not\subseteq L'_{\mathbf{M}'(T[n_i])}$. It follows that $\mathcal{L}'$ has infinite elasticity. A contradiction. $\square$

We next introduce an interesting topological property of a class of languages that is connected to the learnability of the class.

Definition 25 *[24]* L_j *is a* minimal concept of L within $\mathcal{L}$ *just in case* $L \subseteq L_j$, $L_j \in \text{range}(\mathcal{L})$, *and there is no* $L_i \in \text{range}(\mathcal{L})$ *such that* $L \subseteq L_i$ *and* $L_i \subset L_j$.

Definition 26 *[29]* $\mathcal{L}$ *satisfies* MEF-condition *if for each finite set* D *and for each* $L_i \in \text{range}(\mathcal{L})$ *with* $D \subseteq L_i$ *there is a minimal concept* L_j *of* D *within* $\mathcal{L}$ *such that* $L_j \subseteq L_i$. $\mathcal{L}$ *satisfies* MFF-condition *if for any nonempty finite set* D*, the cardinality of* $\{L_i \in \text{range}(\mathcal{L}) \mid L_i$ *is a minimal concept of* D *within* $\mathcal{L}\}$ *is finite.* $\mathcal{L}$ *has* M-finite thickness *if* $\mathcal{L}$ *satisfies both MEF-condition and MFF-condition.*

Theorem 27 *Let* $\mathcal{L} = L_0, L_1, \ldots$ *be an indexed family. Assume that* $\mathcal{L}$ *has M-finite thickness and finite elasticity. Then* $\mathcal{L} \in \mathbf{TxtEx}_\alpha$ *with respect to hypothesis space* $\mathcal{L}$*, for some constructive ordinal notation* α*.*

PROOF. Suppose T is an arbitrary text. We then describe a learning machine $\mathbf{M}$. Define $\mathbf{M}(T[n])$ as follows. Let $L_i^{(n)}$ denote $L_i \cap \{x \mid x < n\}$.

If $\emptyset \in \mathcal{L}$, then let $G_\emptyset$ denote a grammar for $\emptyset$ in $\mathcal{L}$; otherwise let $G_\emptyset = 0$.

$\mathbf{M}(T[n])$

Let $C_n = \text{content}(T[n])$.

If $C_n = \emptyset$ then output $G_\emptyset$.

Let $S_n = \{i \leq n \mid C_n \subseteq L_i \ \wedge \ \neg(\exists j \leq n)[C_n \subseteq L_j \ \wedge \ L_j^{(n)} \subset L_i^{(n)}]\}$.

If S_n is not empty then output $\min(S_n)$, else output $\mathbf{M}(T[n-1])$.

End

The above learning machine is a slight modification of the machine of Mukouchi [24].

Let T be an arbitrary text (for a language L). Assume without loss of generality that $\text{content}(T) \neq \emptyset$. We will show that $\mathbf{M}$ makes only finitely many mind changes on T. Suppose for contradiction, $\mathbf{M}$ changes its mind infinitely often on T. First note that, if $\mathbf{M}(T[n]) \neq \mathbf{M}(T[n+1])$ then $\text{content}(T[n+1]) \subseteq L_{\mathbf{M}(T[n+1])}$. Consider two cases:

Case 1. $\text{card}(\{\mathbf{M}(T[n]) \mid n \in N \wedge \text{content}(T) \not\subseteq L_{\mathbf{M}(T[n])}\}) = \infty$. (That is, $\mathbf{M}$, on T, outputs infinitely many distinct conjectures i such that $\text{content}(T) \not\subseteq L_i$.)

Let $n_1 < n_2 < n_3 < \cdots$ be such that $\mathbf{M}(T[n_i]) \neq \mathbf{M}(T[n_{i+1}])$, and $\text{content}(T[n_{i+1}]) \not\subseteq L_{\mathbf{M}(T[n_i])}$. Note that there exist such an n_i by the hypothesis of this case. Also, by construction, we have $\text{content}(T[n_i]) \subseteq L_{\mathbf{M}(T[n_{i+1}])}$ because any *new* hypothesis output by $\mathbf{M}$ is consistent with the input.

It follows that $\mathcal{L}$ has infinite elasticity (by considering the languages $L_{\mathbf{M}(T[n_{2i}])}$, we see that $\text{content}(T[n_{2i+1}]) \subseteq L_{\mathbf{M}(T[n_{2i+2}])}$, but $\text{content}(T[n_{2i+1}]) \not\subseteq L_{\mathbf{M}(T[n_{2i}])}$.) A contradiction.

Case 2. $\mathbf{M}$, on T, issues only finitely many distinct conjectures i such that $\text{content}(T) \not\subseteq L_i$.

Then, for large enough n, $L_{\mathbf{M}(T[n])} \supseteq \text{content}(T) = L$ (since $\mathbf{M}$ changes its hypothesis infinitely often and if $\mathbf{M}(T[n]) \neq \mathbf{M}(T[n+1])$ then $\text{content}(T[n+1]) \subseteq L_{\mathbf{M}(T[n+1])}$).

Mukouchi [24] showed the following lemma.

Lemma 28 [24] *Let $\mathcal{L} = L_0, L_1, \ldots$ be an indexed family. Let $\mathcal{L}$ satisfy the MEF-condition and have finite elasticity. Let L be a nonempty language. If for some n, $L \subseteq L_n$, then*

(a) there is a minimal concept L_j of L within $\mathcal{L}$ such that $L_j \subseteq L_n$, and

(b) if L_k is a minimal concept of L within $\mathcal{L}$, then there exists a finite $X \subseteq L$ such that L_k is a minimal concept of X within $\mathcal{L}$.

Since, we have already shown that for large enough n, $L_{\mathbf{M}(T[n])} \supseteq L$, Lemma 28 implies that there is a minimal concept of L within $\mathcal{L}$. Let j be the minimal number such that L_j is a minimal concept of L within $\mathcal{L}$. Let $X \subseteq L$ be a finite set such that L_j is a minimal concept of X within $\mathcal{L}$ (by Lemma 28 there exists such an X). Let $S = \{L_k \mid L_k$ is a minimal concept of X within $\mathcal{L}\}$. Note that S is finite, since $\mathcal{L}$ satisfies MFF condition. Let s be so large that for all $L_k \in S$, such that $L_k \neq L_j$, $L_k^{(s)} - L_j^{(s)} \neq \emptyset$ (there exists such an s, since S is finite). Note that this implies, for all k, if $X \subseteq L_k$, then either $L_j \subseteq L_k$, or $L_k^{(s)} - L_j^{(s)} \neq \emptyset$.

Let $m \geq \max(\{s, j\})$, be such that,

(a) $X \subseteq \text{content}(T[m])$,

(b) for all $k < j$, either $\text{content}(T[m]) \not\subseteq L_k$, or $(\exists k' \leq m)[L_{k'} \subset L_k \ \wedge (\exists x \leq m)[L_{k'}^{(x)} \subset L_k^{(x)}]$.

Note that there exists such an m — for part (a), existence of such an m is obvious; for part (b) existence of such an m follows from the fact that none of L_k, $k < j$, is a minimal concept for L within $\mathcal{L}$.

Now suppose $n \geq m$. Consider S_n as defined in $\mathbf{M}(T[n])$. It follows from (b) above that for all $n \geq m$, S_n does not contain any number $< j$. Furthermore, S_n contains j, since for all k such that $X \subseteq L_k$, either $L_k^{(s)} - L_j^{(s)} \neq \emptyset$, or $L_j \subseteq L_k$. It follows that $\mathbf{M}(T)\!\downarrow = j$.

Thus, $\mathbf{M}$ must make only finitely many mind changes on every text T. Similarly to Case 2, we can show that on any text for a language L_j, $\mathbf{M}$ converges to the smallest index for L_j. So, $\mathbf{M}$ makes finitely many mind changes on any input and **TxtEx**-identifies $\mathcal{L}$ with respect to $\mathcal{L}$. Thus, Theorem 17 implies that $\mathcal{L} \in \mathbf{TxtEx}_\alpha$ with respect to $\mathcal{L}$, for some constructive ordinal notation α. □

Corollary 29 *Let $\mathcal{L}$ be an indexed family with finite thickness. Then $\mathcal{L} \in \mathbf{TxtEx}_\alpha$ with respect to $\mathcal{L}$, for some constructive ordinal notation α.*

PROOF. If $\mathcal{L}$ has finite thickness, then $\mathcal{L}$ has finite elasticity (cf. Wright [33] and Shinohara [32]) and M-finite thickness (cf. Mukouchi [24]). Hence, by Theorem 27, $\mathcal{L} \in \mathbf{TxtEx}_\alpha$ with respect to $\mathcal{L}$, for some constructive ordinal notation α. □

A special case of Theorem 27 is the learnability of length-bounded elementary formal systems with ordinal-bounded mind changes. (Shinohara [32] has proved that $LBEFS^{(\leq n)}$, the class of languages defined by length-bounded elementary formal systems with at most n axioms, has finite elasticity and Sato and Moriyama [29] have proved that $LBEFS^{(\leq n)}$ has M-finite thickness.) The learnability of $LBEFS^{(\leq n)}$ was shown by Shinohara [32]. Jain and Sharma [14] proved that $LBEFS^{(\leq n)}$ is learnable with the number of mind changes bounded by ordinal ω^n.

The results discussed in the present paper give general sufficient conditions for identifiability with ordinal bound on mind changes. However, they do not give explicit ordinals α. In all these theorems we have "$\mathcal{L} \in \mathbf{TxtEx}_\alpha$ for some constructive ordinal notation α." It appears that ordinal α can be arbitrarily large. An interesting question to ask is if the ordinal bound α is still arbitrarily large if attention is restricted to classes that are identifiable by strategies that are restricted to obeying monotonicity properties. The next result implies that even if we require that a class $\mathcal{L}$ has finite thickness and that it is identifiable by a strong-monotonic learning machine, the ordinal mind change bound can be arbitrarily large. The reader should however note that strong-monotonicity together with finite thickness implies the existence of an ordinal bound because strong-monotonicity implies conservatism and finite-thickness implies finite elasticity (see [21]).

5 Ordinal Complexity and Monotonicity

Below we describe the notion of strong-monotonic identification.

Definition 30 *[16] Let $\mathcal{L}' = L'_0, L'_1, \ldots$ be a hypothesis space.*

(a) A learning machine $\mathbf{M}$ *is said to be* strong monotonic with respect to $\mathcal{L}'$ *just in case for all σ and τ such that $\sigma \subseteq \tau$, $L'_{\mathbf{M}(\sigma)} \subseteq L'_{\mathbf{M}(\tau)}$.*
(b) A learning machine $\mathbf{M}$ *is said to* strong-monotonically $\mathbf{TxtEx}$-identify L with respect to $\mathcal{L}'$ *just in case* $\mathbf{M}$ $\mathbf{TxtEx}$*-identifies L with respect to $\mathcal{L}'$ and* $\mathbf{M}$ *is strong monotonic with respect to $\mathcal{L}'$.*
(c) $\mathbf{M}$ strong-monotonically $\mathbf{TxtEx}$-identifies $\mathcal{L}$ with respect to $\mathcal{L}'$ *just in case, for each $L \in$* range$(\mathcal{L})$*,* $\mathbf{M}$ *strong-monotonically* $\mathbf{TxtEx}$*-identifies L with respect to $\mathcal{L}'$.*

Theorem 31 *Let α be a constructive ordinal notation. There exists an indexed family $\mathcal{L}$ such that $\mathcal{L}$ can be* $\mathbf{TxtEx}$*-identified strong-monotonically with respect to hypothesis space $\mathcal{L}$, $\mathcal{L}$ has finite thickness, and $\mathcal{L} \notin \mathbf{TxtEx}_\alpha$ with respect to any hypothesis space.*

PROOF. Fix constructive ordinal notation α. Let $(\mathbf{M}_0, \mathbf{F}_0), (\mathbf{M}_1, \mathbf{F}_1) \ldots$ be an enumeration of pairs of learning machines and corresponding ordinal mind change counter functions as given by Lemma 10. Using an argument similar to the one used by [9] for function learning, one can show that, for each $i \in N$, and for each text T, $\mathbf{M}_i$ makes only finitely many mind changes when fed T.

Let $L_i = \{\langle i, x\rangle \mid x \in N\}$. Note that L_i is infinite, and for distinct i, j, L_i and L_j are disjoint. Let $L_i^s = \{\langle i, x\rangle \mid x \leq s\}$. We now give an algorithm which receives i and enumerates (effectively in i) a (finite) sequence [4] $\mathcal{L}_i$ of languages such that:

(a) if $L \in \mathrm{range}(\mathcal{L}_i)$, then $L = L_i^s$ for some s;
(b) $\mathrm{range}(\mathcal{L}_i)$ is finite (note that one can effectively decide the membership problem for languages in $\mathcal{L}_i$);
(c) $\mathrm{range}(\mathcal{L}_i)$ is not **TxtEx**-identified by $\mathbf{M}_i$ with respect to any hypothesis space;
(d) There exists a machine, effective in i, that strong-monotonically **TxtEx**-identifies $\mathrm{range}(\mathcal{L}_i)$ with respect to the hypothesis space $\mathcal{L}_i$.

Now define an indexed family $\mathcal{L}$ with $\mathrm{range}(\mathcal{L}) = \bigcup_{i\in N} \mathrm{range}(\mathcal{L}_i)$, such that for $L_i^s \in \mathrm{range}(\mathcal{L})$, one can effectively, in i and s, find an index (in $\mathcal{L}$) for L_i^s. We will show that $\mathcal{L}$ establishes the theorem. First, the algorithm enumerating $\mathcal{L}_i$ is as follows:

Enumeration of $\mathcal{L}_i$.
Initially, let $\mathcal{L}_i$ consists of just the language L_i^0.
Let $n = 0$ and let σ_0 be the least initial sequence such that $\mathrm{content}(\sigma_0) = L_i^0$. Go to Stage 0.
Stage s.
Add the language L_i^{s+1} to $\mathcal{L}_i$.
Search for a sequence γ extending σ_s, such that $\mathrm{content}(\gamma) \subseteq L_i^{s+1}$, and $\mathbf{M}_i(\sigma_s) \neq \mathbf{M}_i(\gamma)$.
If and when such a γ is found, let σ_{s+1} be the least extension of γ such that $\mathrm{content}(\sigma_{s+1}) = L_i^{s+1}$.
Go to Stage $s+1$.
End Stage s
End Enumeration of $\mathcal{L}_i$

We now show that $\mathcal{L}_i$, $i \in N$, constructed above satisfy the properties claimed.

Lemma 32 *For each $i \in N$, there are only finitely many stages in the enumeration procedure for $\mathcal{L}_i$. Hence,* $\mathrm{range}(\mathcal{L}_i)$ *is finite.*

[4] Strictly speaking, an indexed family is an infinite sequence of languages. For ease of presentation, the algorithm here describes enumeration of only a finite sequence. One can easily obtain an infinite sequence by just repeating the languages in $\mathrm{range}(\mathcal{L}_i)$.

PROOF. Suppose by way of contradiction there is an $i \in N$ such that there are infinitely many stages in the constrution of $\mathcal{L}_i$. Then $\mathbf{M}_i$ on $\bigcup_{s \in N} \sigma_s$ makes infinitely many mind changes, a contradiction. □

Lemma 33 *For each $i \in N$, $\mathbf{M}_i$ fails to* **TxtEx**-*identify* range($\mathcal{L}_i$) *with respect to any hypothesis space.*

PROOF. Let s be the stage in the enumeration of $\mathcal{L}_i$ which starts but does not terminate. Then $\mathbf{M}_i$ can **TxtEx**-identify at most one of L_i^s and L_i^{s+1}, both of which are in range($\mathcal{L}_i$). □

We continue with the proof of the theorem. Now define $\mathcal{L}$ such that range($\mathcal{L}$) = $\bigcup_{i \in N}$ range($\mathcal{L}_i$), and for $L_i^s \in \mathcal{L}$, one can effectively, in i and s, find an index (in $\mathcal{L}$) for L_i^s. It is easy to verify that $\mathcal{L}$ can be strong monotonically identified with respect to hypothesis space $\mathcal{L}$. Also, $\mathcal{L} \notin \mathbf{TxtEx}_\alpha$, by Lemma 33. Moreover, note that L_i's are pairwise disjoint. Thus, since each language in $\mathcal{L}_i$ is a subset of L_i and $\mathcal{L}_i$ is finite, we have that $\mathcal{L}$ has finite thickness. □

The reader should note that a similar result in the sense of class-preserving or exact identification cannot hold for dual strong-monotonicity [17] because class preserving dual strong monotonic identification is the same as finite identification (see [20], [34]). However, we can establish a similar result for class comprising dual strong monotonic identification which is a proper superset of finite identification (see [21]).

Definition 34 *[17] Let $\mathcal{L}' = L'_0, L'_1, \ldots$ be a hypothesis space.*

(a) A learning machine $\mathbf{M}$ *is said to be* dual strong-monotonic with respect to the hypothesis space $\mathcal{L}'$ *just in case for all σ and τ such that $\sigma \subseteq \tau$, $L'_{\mathbf{M}(\sigma)} \supseteq L'_{\mathbf{M}(\tau)}$.*

(b) A learning machine $\mathbf{M}$ *is said to* dual strong-monotonically **TxtEx**-identify L with respect to the hypothesis space $\mathcal{L}'$ *just in case* $\mathbf{M}$ **TxtEx**-*identifies L with respect to the hypothesis space $\mathcal{L}'$ and* $\mathbf{M}$ *is dual strong monotonic with respect to $\mathcal{L}'$.*

(c) $\mathbf{M}$ dual strong-monotonically **TxtEx**-identifies $\mathcal{L}$ with respect to hypothesis space $\mathcal{L}'$ *just in case, for each $L \in$* range($\mathcal{L}$), $\mathbf{M}$ *dual strong-monotonically* **TxtEx**-*identifies L with respect to $\mathcal{L}'$.*

Theorem 35 *Let α be a constructive ordinal notation. There exists an indexed family $\mathcal{L}$ and a hypothesis space $\mathcal{L}'$ such that $\mathcal{L}$ can be* **TxtEx**-*identified* *dual strong-monotonically with respect to $\mathcal{L}'$, $\mathcal{L}'$ has finite thickness, and $\mathcal{L} \notin$* $\mathbf{TxtEx}_\alpha$ *with respect to any hypothesis space.*

PROOF. Fix constructive ordinal notation α. Let $(\mathbf{M}_0, \mathbf{F}_0), (\mathbf{M}_1, \mathbf{F}_1) \ldots$ be an enumeration of pairs of learning machines and corresponding ordinal mind change counter functions as given by Lemma 10. Using an argument similar to the one used by [9] for function learning, one can show that, for each $i \in N$, and for any text T, $\mathbf{M}_i$ makes only finitely many mind changes when fed T.

For each i, we will define a recursive function g_i (where a program for g_i can be found effectively in i). g_i will satisfy the following properties:

(A) $\{x \mid g_i(x) = 1\}$ is nonempty and finite. Moreover, $\{x \mid g_i(x) = 1\} \subseteq \{\langle i, y\rangle \mid y \in N\}$.

(B) Let $L_i = \{2x, 2x+1 \mid g_i(x) = 1\}$. Let $\mathcal{C}_i = \{L \subseteq L_i \mid (\forall x \mid g_i(x) = 1)(\exists! b \in \{0,1\})[2x + b \in L]\}$. Then, $\mathcal{C}_i \not\subseteq \mathbf{TxtEx}_\alpha(\mathbf{M}_i, \mathbf{F}_i)$ (with respect to any hypothesis space) [5] .

We take $\mathcal{L}$ to be an indexed family such that range$(\mathcal{L}) = \bigcup_i \mathcal{C}_i$ (using the fact that $g_i^{-1}(1)$ is finite, one can easily construct such an indexed family $\mathcal{L}$). From (B) it follows that $\mathcal{L} \notin \mathbf{TxtEx}_\alpha$ with respect to any hypothesis space.

We let $\mathcal{L}'$ be an hypothesis space such that range$(\mathcal{L}') = \{L \mid (\exists i)[L \subseteq L_i]\}$, where an index for $L_i - D$, for any finite set D, can be obtained effectively from i and D. Note that such an hypothesis space $\mathcal{L}'$ can be easily constructed. Clearly, $\mathcal{L}'$ has finite thickness.

It remains to construct recursive functions g_i as claimed above and to show that $\mathcal{L}$ can be dual strong monotonically identified with respect to hypothesis space $\mathcal{L}'$.

We now define g_i.

Definition of g_i

For $x < \langle i, 0\rangle$, let $g_i(x) = 0$. Let $g_i(\langle i, 0\rangle) = 1$.

Let $x_i^0 = \langle i, 0\rangle$. Intuitively, x_i^s denotes the largest x such that $g_i(x)$ is defined to be 1 before stage s.

Let $\sigma_0 = \Lambda$.

Go to Stage 0.

Stage s

1. Dovetail steps 2 and 3, until step 2 succeeds. If and when step 2 succeeds, go to step 4.

[5] *Notation*: $\exists!$ denotes "there exists a unique."

2. Search for an extension τ of σ_s, and $z \in \{2x_i^s, 2x_i^s+1\}$ such that
   (a) $\mathbf{M}_i(\tau) \neq \mathbf{M}_i(\sigma_s)$, and
   (b) $\text{content}(\tau) = \text{content}(\sigma_s) \cup \{z\}$.
3. For $x = x_i^s + 1$ to ∞ do
   Let $g_i(x) = 0$.
   EndFor
4. If and when such τ, z are found, let $\sigma_{s+1} = \tau$. Let $x_i^{s+1} \in \{\langle i, y\rangle \mid y \in N\}$ be the least number such that $g_i(x_i^{s+1})$ has not been defined until now.
   Let $g_i(x_i^{s+1}) = 1$.
   For $x < x_i^{s+1}$ such that $g_i(x)$ has not been defined until now, let $g_i(x) = 0$.

End Stage s

End of definition of g_i.

Lemma 36 *For each $i \in N$, there are only finitely many stages in the construction of g_i.*

PROOF. Suppose by way of contradiction there are infinitely many stages. Then, $\mathbf{M}_i$ on $\bigcup_{s\in N} \sigma_s$ makes infinitely many mind changes, a contradiction. □

We continue with the proof of the theorem. Fix i. Using the above lemma, it is easy to verify that g_i satisfies (A). We now show that g_i satisfies (B). Suppose s is the stage which starts but does not terminate. Let $L' = \text{content}(\sigma_s) \cup \{2x_i^s\}$. Let $L'' = \text{content}(\sigma_s) \cup \{2x_i^s+1\}$. Let T', extending σ_s, be a text for L'. Let T'' extending σ_s be a text for L''. Since step 2 in stage s did not succeed, we have that $\mathbf{M}_i(T') = \mathbf{M}_i(T'') = \mathbf{M}_i(\sigma_s)$. It follows that $\mathbf{M}_i$ does not **TxtEx**-identify $\mathcal{L}_i$ with respect to any hypothesis space. Thus, (B) is satisfied.

We now give a machine $\mathbf{M}$ which, for each $L \in \mathcal{L}$, dual strong monotonically identifies L with respect to hypothesis space $\mathcal{L}'$. Let gram be a recursive function such that $L'_{\text{gram}(i,D)} = L_i - D$ (by construction of $\mathcal{L}'$ such a function gram clearly exists).

For $x \in N$ and $b \in \{0,1\}$, let $\text{mate}(2x+b) = 2x+1-b$.

$\mathbf{M}(T[n])$

If $\text{content}(T[n]) = \emptyset$, then let $\mathbf{M}(T[n]) =?$.

1. Let i be such that $\text{content}(T[n]) \subseteq \{2\langle i, y\rangle + b \mid y \in N \ \wedge \ b \in \{0,1\}\}$.
   (If no such i exists, then let $\mathbf{M}(T[n]) = \mathbf{M}(T[n-1])$.)
2. Let $D = \{\text{mate}(z) \mid z \in \text{content}(T[n])\}$.
3. Output $\text{gram}(i, D)$.

End

It is easy to verify from the definition of L_i, $\mathcal{C}_i$, $\mathcal{L}$, $\mathcal{L}'$ that $\mathbf{M}$ is dual strong monotonic and **TxtEx**-identifies $\mathcal{L}$ with respect to hypothesis space $\mathcal{L}'$. The theorem follows. $\square$

6 Conclusion

The present paper further illustrated the utility of ordinal mind change bound as a measure of the difficulty of learning a class of languages. From the ordinal mind change complexity results for bounded unions of pattern languages, it is clear that the presence of negative data in addition to positive data makes the learning task much simpler. The ordinal bounds, in some sense, give a measure of "how much simpler." It was argued that the existence of an ordinal mind change bound can be viewed as a measure of learning "tractability." Several sufficient conditions were derived for the existence of such a bound in terms of various topological properties of language classes.

The techniques presented in the paper yield a useful measure to compare the complexity of learning of rich classes of concepts which are not very amenable to analysis by more restricted notions of complexity. This is especially true of concept classes that go beyond propositional representation, e.g., elementary formal systems and logic programming systems. For the classes of languages considered in the present paper, only negative learnability results are possible with more restricted models like PAC. For example, the class of pattern languages is not PAC learnable even if both positive and negative data are available (see Schapire [30]). Hence, models like PAC appear to be too restrictive for analyzing learning complexity of unions of pattern langauges or elementary formal systems. The ordinal mind change complexity model considered in the present paper gives a measure of the mind change complexity that a learner makes in learning these classes. At present, this appears to be one of the very few models that quantitatively analyzes the learning difficulty of such expressive languages. Other models that attempt the address the complexity of identification in the limit are due to Daley and Smith [8] and due to Pitt [26].

Acknowledgement

We thank the referees for several helpful comments that have improved the paper. Andris Ambainis was supported in part by Latvia's Science Council

Grants 93.599 and 96.0282 to R. Freivalds. Arun Sharma was supported in part by the Australian Research Council Grants A49600456 and A49803051.

References

[1] A. Ambainis. Power of procrastination in inductive inference: How it depends on used ordinal notations. In Paul Vitányi, editor, *Second European Conference on Computational Learning Theory*, volume 904 of *Lecture Notes in Artificial Intelligence*, pages 99–111. Springer-Verlag, 1995.

[2] D. Angluin. Finding patterns common to a set of strings. *Journal of Computer and System Sciences*, 21:46–62, 1980.

[3] D. Angluin. Inductive inference of formal languages from positive data. *Information and Control*, 45:117–135, 1980.

[4] S. Arikawa, S. Miyano, A. Shinohara, T. Shinohara, and A. Yamamoto. Algorithmic learning theory with elementary formal systems. *IEICE Trans. Inf. and Syst.*, E75–D No. 4:405–414, 1992.

[5] J. Bārzdiņš and K. Podnieks. The theory of inductive inference. In *Second Symposium on Mathematical Foundations of Computer Science*, pages 9–15. Math. Inst. of the Slovak Academy of Sciences, 1973.

[6] J. Case, S. Jain, and M. Suraj. Not-so-nearly-minimal-size program inference. In K. Jantke and S. Lange, editors, *Algorithmic Learning for Knowledge-Based Systems*, volume 961 of *Lecture Notes in Artificial Intelligence*, pages 77–96. Springer-Verlag, 1995.

[7] J. Case and C. Smith. Comparison of identification criteria for machine inductive inference. *Theoretical Computer Science*, 25:193–220, 1983.

[8] R. Daley and C. Smith. On the complexity of inductive inference. *Information and Control*, 69:12–40, 1986.

[9] R. Freivalds and C. Smith. On the role of procrastination in machine learning. *Information and Computation*, pages 237–271, 1993.

[10] E. M. Gold. Language identification in the limit. *Information and Control*, 10:447–474, 1967.

[11] S. Jain and A. Sharma. On the intrinsic complexity of language identification. In *Proceedings of the Seventh Annual Conference on Computational Learning Theory*, pages 278–286. ACM-Press, 1994.

[12] S. Jain and A. Sharma. Elementary formal systems, intrinsic complexity, and procrastination. In *Proceedings of the Ninth Annual Conference on Computational Learning Theory*, pages 181–192. ACM-Press, 1996.

[13] S. Jain and A. Sharma. The intrinsic complexity of language identification. *Journal of Computer and System Sciences*, 52:393–402, 1996.

[14] S. Jain and A. Sharma. Elementary formal systems, intrinsic complexity, and procrastination. *Information and Computation*, 132:65–84, 1997.

[15] S. Jain and A. Sharma. The structure of intrinsic complexity of learning. *Journal of Symbolic Logic*, 62:1187–1201, 1997.

[16] K. Jantke. Monotonic and non-monotonic inductive inference. *New Generation Computing*, 8:349–360, 1991.

[17] S. Kapur. Monotonic language learning. In S. Doshita, K. Furukawa, K. Jantke, and T. Nishida, editors, *Algorithmic Learning Theory: Third International Workshop (ALT '92)*, volume 743 of *Lecture Notes in Artificial Intelligence*, pages 147–158. Springer-Verlag, 1992.

[18] P. Kilpeläinen, H. Mannila, and E. Ukkonen. MDL learning of unions of simple pattern languages from positive examples. In Paul Vitányi, editor, *Second European Conference on Computational Learning Theory*, volume 904 of *Lecture Notes in Artificial Intelligence*, pages 252–260. Springer-Verlag, 1995.

[19] S. Kleene. Notations for ordinal numbers. *Journal of Symbolic Logic*, 3:150–155, 1938.

[20] S. Lange and T. Zeugmann. Monotonic versus non-monotonic language learning. In *Proceedings of the Second International Workshop on Nonmonotonic and Inductive Logic*, volume 659 of *Lecture Notes in Artificial Intelligence*, pages 254–269. Springer-Verlag, 1993.

[21] S. Lange, T. Zeugmann, and S. Kapur. Monotonic and dual monotonic language learning. *Theoretical Computer Science A*, 155:365–410, 1996.

[22] M. Machtey and P. Young. *An Introduction to the General Theory of Algorithms*. North Holland, New York, 1978.

[23] T. Motoki, T. Shinohara, and K. Wright. The correct definition of finite elasticity: Corrigendum to identification of unions. In L. Valiant and M. Warmuth, editors, *Proceedings of the Fourth Annual Workshop on Computational Learning Theory*, page 375. Morgan Kaufmann, 1991.

[24] Y. Mukouchi. Inductive inference of an approximate concept from positive data. In S. Arikawa and K. Jantke, editors, *Algorithmic learning theory: Fourth International Workshop on Analogical and Inductive Inference (AII '94) and Fifth International Workshop on Algorithmic Learning Theory (ALT '94)*, volume 872 of *Lecture Notes in Artificial Intelligence*, pages 484–499. Springer-Verlag, 1994.

[25] D. Osherson, M. Stob, and S. Weinstein. *Systems that Learn: An Introduction to Learning Theory for Cognitive and Computer Scientists*. MIT Press, 1986.

[26] L. Pitt. Inductive Inference, DFAs, and Computational Complexity. In *Analogical and Inductive Inference, Proceedings of the Second International Workshop (AII '89)*, volume 397 of *Lecture Notes in Artificial Intelligence*, pages 18–44. Springer-Verlag, 1989.

[27] H. Rogers. *Theory of Recursive Functions and Effective Computability.* McGraw-Hill, 1967. Reprinted, MIT Press 1987.

[28] G. Sacks. *Higher Recursion Theory.* Springer-Verlag, 1990.

[29] M. Sato and T. Moriyama. Inductive inference of length bounded EFS's from positive data. Technical Report DMSIS-RR-94-2, Department of Mathematical Sciences and Information Sciences, University of Osaka Prefecture, Japan, 1994.

[30] R.E. Schapire. Pattern languages are not learnable. In M. Fulk and J. Case, editors, *Proceedings of the Third Annual Workshop on Computational Learning Theory*, pages 122–129. Morgan Kaufmann, 1990.

[31] T. Shinohara. *Studies on Inductive Inference from Positive Data.* PhD thesis, Kyushu University, Kyushu, Japan, 1986.

[32] T. Shinohara. Rich classes inferable from positive data: Length–bounded elementary formal systems. *Information and Computation*, 108:175–186, 1994.

[33] K. Wright. Identification of unions of languages drawn from an identifiable class. In R. Rivest, D. Haussler, and M. Warmuth, editors, *Proceedings of the Second Annual Workshop on Computational Learning Theory*, pages 328–333. Morgan Kaufmann, 1989.

[34] T. Zeugmann and S. Lange. A guided tour across the boundaries of learning recursive languages. In K. Jantke and S. Lange, editors, *Algorithmic Learning for Knowledge-Based Systems*, volume 961 of *Lecture Notes in Artificial Intelligence*, pages 190–258. Springer-Verlag, 1995.